Vicinity Characterization of Monotone Non-Degenerate Boolean Functions

Patrícia Tenera Roxo
Instituto Superior Técnico, Lisboa, Portugal

December 2023

Abstract

The goal of the thesis is to characterize the immediate vicinity of a given monotone non-degenerate Boolean function within that same set of monotone non-degenerate Boolean functions. The underlying motivation for this restriction is the use of these functions in the modeling of Biological Regulatory Networks.

The results established in this work capitalize on the unique representation of monotone non-degenerate Boolean functions by means of their Complete Disjunctive Normal Form (CDNF). From their CDNF, non-degenerate monotone Boolean functions can be embedded into sets by fixing the sign of each variable appearing in the function in a structure called a signature. The characterization of all immediate relatives of a given function is split between the characterization of immediate relatives of the function belonging to the same signature as that of the given function, and the characterization of immediate relatives of the function belonging to a distinct signature as that of the given function.

This approach of fixing a target signature in which to look for immediate relatives of the desired function has significant benefits for the background motivational problem.

Keywords: Monotone Non-Degenerate Boolean Functions, Partial Order, Complete Disjunctive Normal Form, Set Representation, Vicinity Characterization.

1. Introduction

Boolean functions are one of the most studied Mathematical objects, in part due to their conceptual simplicity but mostly due to their exceptionally broad spectrum of applications. Not only do Boolean functions play a key role in Optimization and Combinatorics, Coding Theory, Propositional Satisfiability Solving, but in real-world applications as well, such as in Circuit design, Cryptography, and in particular, Biology.

There are many ways to model processes in Biology, but these models fall roughly into two categories: qualitative and quantitative models [1, 2]. Under qualitative models, Boolean Networks are of particular prominence. Their significance lies not only in their proven success in modeling Regulatory Networks [3–6], but also in their ability to provide valuable insights in assessing a proposed model’s robustness [7] as well as in model revision [8, 9]. It is in this context of model revision that the work presented in this thesis arises.

Suppose a domain expert proposes an initial Boolean function to model some process of interest, but this model is deemed inconsistent with some newly collected experimental data. Naturally, the question arises: How can one repair the model to make it consistent with both the previously and the newly collected data?

2. Background

Definition 2.1. For a given Boolean function, \( f \), let \( \mathcal{T}(f) \) denote the True set of \( f \). That is, \( \mathcal{T}(f) \) is the set of inputs that make \( f \) yield True.

Definition 2.2. Let \( \mathcal{F}_n \) denote the set of all monotone non-degenerate Boolean functions over \( n \) variables.

Definition 2.3. Let \( f, f' \in \mathcal{F}_n \). Then \( f \preceq f' \) whenever \( \mathcal{T}(f) \subseteq \mathcal{T}(f') \).

Definition 2.4. Let \( c \) and \( c' \) denote two distinct clauses. If \( c' \) is True whenever \( c \) is True, then \( c \) is said to be an impllicant for \( c' \), or conversely, \( c' \) is said to be implied by \( c \).

Definition 2.5. Let \( f \) be a Boolean function, and let \( c \) be a single clause. \( c \) is said to be an implicant for \( f \), if whenever \( c \) is True, \( f \) is also necessarily True.

Definition 2.6. Let \( f \) be a Boolean function, and let \( c \) be a single clause. \( c \) is said to be independent of \( f \) if \( c \) is not an implicant for any clause in \( f \) and if conversely, no clause in \( f \) is an implicant for clause \( c \).

Definition 2.7. Let \( f \) be a Boolean function, and let \( c \) be a single clause. \( c \) is said to be a maximal independent clause of \( f \) if \( c \) is independent of \( f \) and if \( c \) cannot be extended with any literal without becoming implied by some clause in \( f \).
Definition 2.8. Let \( f \) be a Boolean function and let \( c \) be an implicant of \( f \). Clause \( c \) is said to be a prime implicant for \( f \) if no other implicant of \( f \), \( c' \), exists such that \( c' \) is also an implicant of \( c \).

Definition 2.9. Let \( f \) be a Boolean function, and let \( c \) be a single clause. Clause \( c \) is said to maximal dominated by \( f \) if \( c \) is a prime implicant for some clause in \( f \).

Definition 2.10. The Complete Disjunctive Normal Form (CDNF), i.e., Blake Canonical Normal Form, of a Boolean function \( f \) is the disjunction of all prime implicants of \( f \).

Theorem 2.1. Let \( f \) be a non-degenerate monotone Boolean function. Then, \( f \) has a unique prime representation given by its CDNF.

Definition 2.11. Let \( f \) be any non-degenerate monotone Boolean function over \( n \) variables. Given its CDNF \( f = \bigvee_{i=1}^{\ell} c_i \), where each clause \( c_i \) in \( f \) is of the form \( c_i = \bigwedge_{j=1}^{m_i} l_{ij} \), for some literals \( l_{ij} \); then \( \Phi_f = \{ C_i : i = 1, \ldots, k \} \) with \( C_i = \{ I(l_{ij}) : l_{ij} \in c_i \} \), is said to be the set representation of \( f \), where \( I \) is the map that returns the variable index associated with literal \( l_{ij} \).

3. Problem Statement

"Let \( f \) be a given non-degenerate monotone Boolean function over \( n \) variables. Find all other non-degenerate monotone Boolean functions \( f' \in \mathcal{F}_n \) over the same variables such that:

1. \( f \preccurlyeq f' \) and \( \exists f'' \), another non-degenerate monotone Boolean function such that \( f \preccurlyeq f'' \preccurlyeq f' \)
2. \( f' \preccurlyeq f \) and \( \exists f'' \), another non-degenerate monotone Boolean function such that \( f' \preccurlyeq f'' \preccurlyeq f' \)

The functions satisfying the first conditions are called immediate parents of \( f \), whereas the functions satisfying the second conditions are called immediate children of \( f \).

The significance of computing these sets lies in the observation that if \( f' \) is an immediate parent of \( f \), then \( f' \) is True whenever \( f \) is True. In the context of the motivating Biological problem at hand, this means that if the initially proposed model \( f \) produces False when experience claims it should yield True, one needs only navigate 'upwards' through the parents of \( f \) to find a new model, \( f' \), consistent with the previously inconsistent input. Similarly, if experience suggests the model should output False and True is obtained, then one needs only navigate 'downwards' through the children of \( f \).

4. Signature Approach

In order to utilize the unique representation provided by the CDNF of a non-degenerate monotone Boolean function, one must store the information regarding the signs of each variable appearing in the given function somewhere. This somewhere is called a signature.

Definition 4.1. \( \Sigma \) is said to be a signature for variables \( \{x_1, \ldots, x_n\} \) whenever \( \Sigma \) is a map \( \Sigma : \{1, \ldots, n\} \rightarrow \mathbb{B} \).

By convention, it is established that if \( \Sigma(i) = 1 \), \( x_i \) only appears in its non-complemented version and conversely, if \( \Sigma(i) = 0 \), \( x_i \) only appears in its complemented version. In order to answer the more general problem as stated above, the thesis mostly focuses on the following simplified problem:

"Let \( f \) be a given non-degenerate monotone Boolean function over \( n \) variables, belonging to some signature \( \Sigma \). Fixing some signature \( \Sigma' \), not necessarily distinct from \( \Sigma \), find all other non-degenerate monotone Boolean functions \( f' \in \Sigma' \) over the same variables such that:

1. \( f \preccurlyeq f' \) and \( \exists f'' \) \( \in \Sigma' \), another non-degenerate monotone Boolean function such that \( f \preccurlyeq f'' \preccurlyeq f' \)
2. \( f' \preccurlyeq f \) and \( \exists f'' \) \( \in \Sigma' \), another non-degenerate monotone Boolean function such that \( f' \preccurlyeq f'' \preccurlyeq f' \)

By answering this second problem, the first problem can also be answered by noticing that if \( f' \), an element of \( \Sigma' \), is an immediate parent (respectively, immediate child) of \( f \) in \( \mathcal{F}_n \), then \( f' \) is an immediate parent (respectively, immediate child) of \( f \) within its own signature, \( \Sigma' \). Of course this approach is not without its challenges, which are discussed at the end of Chapter 4 of the thesis.

The benefit of adding this new layer of restrictions is two-fold: not only is it mathematically easier to tackle, but it also has contextual Biological advantages. Fixing the sign of a variable roughly translates to identifying its representing component as acting either as an inhibitor or as an activator. By searching for immediate parents (respectively, immediate children) of \( f \) over all of \( \mathcal{F}_n \), one allows for all variables to change signs with respect to the initially proposed model. This may be undesirable. One may want to consider allowing at most some \( k \) variables (fixed or arbitrary) to switch signs, or conversely, one may want...
to assert the hard constraint that some \( k' \) variables (fixed or arbitrary) cannot change signs under any circumstances. By fixing a target signature upfront, both of these requirements can be met.

The thesis' contributions can be classified between immediate parents results, immediate children results, same signature results and distinct signature results. Each theorem has a proposed algorithmic counterpart, for which efficiency is estimated. In the case of searching and computing immediate relatives of a given \( f \) in a signature distinct from the one \( f \) belongs to, existence criteria are provided.

5. Same Signature Results

5.1. Immediate Parents

Theorem 5.1. (Shape of an Immediate Parent of \( f \) within the original signature) Let \( f \) and \( f' \) be elements of \( \mathcal{F}_n \) over a given signature \( \Sigma \). Then \( f \preceq f' \) immediately if and only if \( f' \) is of one of the following forms:

1. \( f' = f \cup \{ c \} \), where \( c \) is a maximal independent clause of \( f \).
2. \( f' = f^* \cup \{ s^* \} \), where \( s^* \) is a maximal dominated clause of \( f \) such that \( s^* \) is not contained in any maximal independent clause of \( f \) and \( f^* \) is obtained from \( f \) by removing all clauses \( s \) such that \( s^* \subseteq s \).
3. \( f' = f^* \cup \{ s^* \} \cup \{ d \} \), where:
   - \( s^* \) and \( d \) are maximal subclauses dominated by the same clause in \( f \);
   - \( s^* \) and \( d \) absorb exactly one clause in \( f \) - the clause that originates them;
   - neither \( s^* \) nor \( d \) are contained in any maximal independent clause of \( f \);
   - neither \( f^* \cup \{ s^* \} \) nor \( f^* \cup \{ d \} \) alone yield a cover, but \( f^* \cup \{ s^* \} \cup \{ d \} \) does.

Parents of the first proposed form are called parents of \( f \) of \( \text{the first kind} \), whereas parents of the second and third form are called parents of \( f \) \( \text{of the second kind} \). The reason behind this being the fact that parents of the \( \text{first kind} \) contain \( f \) intact within them, that is \( \forall s \in f : s \in f' \); whereas parents of the \( \text{second kind} \) do not. This is a key idea throughout the thesis for both parents and children alike: can \( f \)’s structure be preserved or must it necessarily be modified?

Example 5.1. Let \( f = (x_1 \land x_2 \land x_3) \lor (x_4 \land x_5) \) belonging to \( \Sigma \), the signature that fixes every variable in its non-complemented version. Having fixed \( \Sigma \), the set representation of \( f \) is thus \( f = \{1, 2, 3\}, \{4, 5\} \). The maximal independent clauses of \( f \) are \( \{1, 2, 4\}, \{1, 2, 5\}, \{1, 3, 4\}, \{1, 3, 5\}, \{2, 3, 4\} \) and \( \{2, 3, 5\} \), thus theorem 5.1 imposes each immediate parent of \( f \) to be of the first kind and made up of \( f \) together with exactly one of these clauses.

Example 5.2. Let \( \Sigma \) be as before, but take \( f \) to be \( f = \{\{1, 2, 3\}, \{2, 3, 4\}, \{1, 3, 4\}\} \). \( f \) only admits the maximal independent clause \( c = \{1, 2, 4\} \), thus according to theorem 5.1, \( f \) has at least one immediate parent, \( f' = f \cup \{1, 2, 4\} \).

The maximal dominated clauses \( \{1, 3\}, \{2, 3\} \) and \( \{3, 4\} \) are such that they are not contained in any maximal independent clauses of \( f \) and such that upon introducing each of these clauses in \( f \), it does not fail to yield a cover, therefore, each of these clauses can play the role of clause \( s^* \) in theorem 5.1. In turn, this implies that \( f \) also has three other immediate parents: \( \{\{1, 3\}, \{2, 3, 4\}\}, \{\{2, 3\}, \{1, 3, 4\}\} \) and \( \{\{3, 4\}, \{1, 3, 4\}\} \), which are of the second form. As \( f \) has no more maximal dominated clauses that are not contained in \( c \), \( f \) does not have any immediate parents of the third form.

Note that this example shows that a given \( f \) can have immediate parents of multiple kinds simultaneously.

5.2. Immediate Children

Theorem 5.2. (Shape of an Immediate Child of \( f \) within the original signature) Let \( f \) and \( f' \) be elements of \( \mathcal{F}_n \) over a given signature \( \Sigma \). Then \( f' \preceq f \) immediately if and only if \( f' \) is of the following forms:

1. \( f' = \{ s_1 \} \cup \{ s_2 \} \cup \ldots \cup \{ (s_i \cup x_i) \} \cup \ldots \cup \{ (s_i \cup x_i) \cup (s_{i+1} \cup \ldots \cup \{ s_k \} \), where:
   - \( f = \{ s_1 \} \cup \ldots \cup \{ s_k \} \);
   - \( \{ x_{i_1}, \ldots, x_{i_j} \} \) are the literals not in \( s_i \) such that \( \exists x \neq i : s_i \subseteq (s_i \cup \{ x_{i_j} \} \).
2. \( f' = (f \setminus \{ s \}) \), where:
   - \( s \in f \) cannot be extended with any literal without absorbing some other clause in \( f \);
   - \( (f \setminus \{ s \}) \) yields a cover.
3. \( f' = (f \setminus \{ s_i, s_j \}) \cup \{ s' \} \), where:
   - \( s_i, s_j \in f \) and neither can be extended without absorbing some clause other than itself in \( f \);
   - neither \( (f \setminus \{ s_i \}) \), nor \( (f \setminus \{ s_j \}) \) yield a cover;
   - \( s_j \) is such that \( \exists x_j \notin s_j : s_j \subseteq (s_j \cup \{ x_j \}) \) and no other clause in \( f \) other than \( s_j \) and \( s_j \) absorbs \( (s_j \cup \{ x_j \}) =: s' \).
For simplicity we may sometimes denote \( s' \) by \( \{ s_i \} \cup \{ s_j \} \).

Once again, children of the first proposed form are called children of \( f \) \( \text{of the first kind} \), whereas children of the
second and third proposed form are called children of the second kind. The reason behind this partition is the fact that children of the first kind do not force the removal of any clause from \( f \), whereas children of the second kind require either for a clause to be removed, or for a pair of clauses to be merged together.

**Example 5.3.** Recover \( \Sigma \) and \( f = \{ \{1, 2, 3\}, \{4, 5\} \} \) from example 5.1 and take the immediate parent \( f' = \{ \{1, 2, 3\}, \{4, 5\}, \{2, 3, 4\} \} \). Clauses \( \{1, 2, 3\} \) and \( \{4, 5\} \) in \( f' \) can be extended with at least one literal without causing absorption within \( f' \), thus \( f' \) has the immediate children \( \{\{1, 2, 3, 5\}, \{4, 5\}, \{2, 3, 4\}\} \) and \( \{\{1, 2, 3\}, \{4, 5\}, \{2, 4, 5\}, \{3, 4, 5\}, \{2, 3, 4\}\} \), which are of the first kind. Clause \( \{2, 3, 4\} \) cannot be extended with any literal without absorbing some clause in \( f' \), however it can be safely removed while maintaining the cover property, thus \( f' \) also has the immediate child \( \{\{1, 2, 3\}, \{4, 5\}\} \equiv f \), which is of the second kind. As no two other clauses remain, \( f' \) has no more immediate children.

The above example illustrates two important points: (1) a given function can have immediate children of multiple kinds simultaneously and (2) if \( f \nRightarrow f' \) are such that \( f' \) is an immediate parent of \( f \) obtained according to theorem 5.1, then \( f \) is necessarily recovered as an immediate child of \( f' \) according to theorem 5.2.

6. Distinct Signature Results

Unlikely when working within the same signature as the given function, where one needs only rearrange the same building blocks - the agreed upon literals - to find immediate relatives of the given function, when dealing with an arbitrary but distinct signature, some care must be taken in order to ensure that the relationship given in definition 2.3 still holds, despite possible variable sign disagreements.

The challenge when considering signatures other than the one the given function belongs to is ascertaining a way to compare functions belonging to multiple signatures. The first step towards this goal is to establish criterions to decide whether a given target signature can or cannot contain any immediate parents of \( f \) in it (Property 6.1), and whether a given target signature can or cannot contain any immediate children of \( f \) in it (Property 6.3). Definitions 6.1 and 6.2 are essential towards this end.

**Definition 6.1.** Let \( f \) be an element of \( \Sigma_k \) and let \( \Sigma_2 \) be a signature that disagrees with \( \Sigma_1 \) on \( \{x_i, \ldots, x_k\} \). Then \( f^* \) denotes the formula obtained from \( f \) by deleting variables \( \{x_i, \ldots, x_k\} \).

**Definition 6.2.** Let \( f \in \Sigma_1 \) and let \( \Sigma_2 \) be some other signature. Then \( \phi(f) \) denotes the Boolean function obtained from \( f \) by changing the sign of each variable the signatures disagree on to be in accordance with signature \( \Sigma_2 \).

### 6.1. Immediate Parents

**Property 6.1. (Existence of Parent Criterion)**

Let \( \Sigma_1 \) and \( \Sigma_2 \) be two distinct signatures over \( x_1, \ldots, x_n \), such that they disagree on exactly \( k \) variables, \( \{x_i, \ldots, x_k\} \), and let \( f \in \mathcal{F}_n \) belong to \( \Sigma_1 \). Then, there exists some \( f' \) belonging to \( \Sigma_2 \), such that \( f \nRightarrow f' \) if and only if deleting the variables \( \{x_i, \ldots, x_k\} \) from \( f \), does not cause any clause in \( f \) to become empty.

That is, in order for a parent of \( f \) to exist in some target signature \( \Sigma_2 \), differing from \( \Sigma_1 \) on \( \{x_i, \ldots, x_k\} \), then no subset of \( \{x_1, \ldots, x_k\} \) can appear together as a clause of \( f \).

**Theorem 6.2. (Shape of an Immediate Parent of \( f \) in a distinct signature)**

Let \( \Sigma_1 \) and \( \Sigma_2 \) be two distinct signatures over \( \{x_1, \ldots, x_n\} \), such that they disagree on exactly \( k \) variables, \( \{x_i, \ldots, x_k\} \), and let \( f \in \mathcal{F}_n \) belong to \( \Sigma_1 \).

If there exists some \( f' \) in \( \Sigma_2 \), an immediate parent of \( f \), then \( f' \) is of the form:

\[
f' = f^* \cup \{\sigma\}
\]

where,

1. \( f^* \) is obtained from \( f \) by deleting the variables \( \{x_i, \ldots, x_k\} \).
2. \( \sigma \) is any maximal independent clause of \( f^* \) in \( \Sigma_2 \). Note that \( \sigma \) necessarily contains all variables \( \{x_i, \ldots, x_k\} \) as fixed by \( \Sigma_2 \), as well as all variables in \( \Sigma_1 \) and \( \Sigma_2 \) agree on that might be missing in \( f^* \).

Intuitively, the fact that \( f \) only has one possible form of immediate parent in a distinct signature stems from the fact that at least one variable has to switch signs and thus at least one clause in \( f \) is necessarily modified. That is, it is intrinsically impossible to preserve \( f \) integrally in any parent of \( f \) in another
signature, immediate or otherwise. In a way, this provides some degree of freedom with which to build an immediate parent of \( f \) in this target signature. The uniqueness of the proposed form for an immediate parent then follows by a maximality argument. A similar phenomena is verified for immediate children.

**Example 6.1.** Let \( \Sigma_1 \) be the signature that fixes all variables in their non-complemented version, let \( \Sigma_2 \) disagree with \( \Sigma_1 \) on \( x_2, x_3 \) and \( x_7 \) and take \( f = (x_1 \land x_2 \land x_3 \land x_4) \lor (x_1 \land x_2 \land x_3 \land x_5) \lor (x_5 \land x_6 \land x_7) \).

Deleting \( x_2, x_3 \) and \( x_7 \) from \( f \) yields \( f_* = (x_1 \land x_4) \lor (x_1 \land x_5) \lor (x_5 \land x_6) \). Theorem 6.2 now enforces that all immediate parents of \( f \) in \( \Sigma_2 \) are of the form \( f' = f_* \cup \{ \sigma \} \), where \( \sigma \) is a maximal independent clause of \( f_* \) in \( \Sigma_2 \). These clauses are: \((x_1, x_2, x_3, x_6, x_7)\), \((x_2, x_3, x_4, x_5, x_7)\) and \((x_3, x_4, x_5, x_6, x_7)\); thus \( f \) has three immediate parents in \( \Sigma_2 \). Note that the added clause \( \sigma \) always guarantees that the newly constructed parent is a cover of \( \{x_1, \ldots, x_n\} \).

6.2. Immediate Children

**Property 6.3.** (Existence of Child Criterion)

Let \( \Sigma_1 \) and \( \Sigma_2 \) be two distinct signatures over \( \{x_1, \ldots, x_n\} \), such that they disagree on exactly \( k \) variables, \( \{x_{i_1}, \ldots, x_{i_k}\} \), and let \( f \in F_n \) belong to \( \Sigma_1 \). Then, there exists some \( f' \) belonging to \( \Sigma_2 \), such that \( f' \prec f \) if \( f \) and only if there is an invariant clause of \( f \) in \( \phi(f) \subseteq \Sigma_2 \), where \( \phi(f) \) is defined in 6.2.

That is, there must be at least one clause in \( f \) that is unchanged when the signs of the variables in \( \{x_{i_1}, \ldots, x_{i_k}\} \) are flipped to be in accordance with \( \Sigma_2 \). These invariant clauses are called *pivotal clauses*, since the constructed immediate children of \( f \) in \( \Sigma_2 \) pivot on them create a valid formula in \( \Sigma_2 \).

**Theorem 6.4.** (Shape of an Immediate Child of \( f \) in a distinct signature)

Let \( \Sigma_1 \) and \( \Sigma_2 \) be two distinct signatures over \( \{x_1, \ldots, x_n\} \) such that they disagree on exactly \( k \) variables \( \{x_{i_1}, \ldots, x_{i_k}\} \) and let \( f \) be an element of \( \Sigma_1 \).

If there exists some \( f' \in \Sigma_2 \), an immediate child of \( f \), then \( f' \) is of the following form:

\[
f' = \{s_{j_1}\} \cup \{s_{j_2}\} \cup \cdots \cup \{s_{j_r} \cup \{l_{j_{r+1}}\}\} \cup \cdots \cup \{s_{j_r} \cup \{l_{j_{rm}}\}\} \cup \{s_{j_{r+1}}\} \cdots \cup \{s_{j_l}\},
\]

where:

- the clauses \( s_{j_i} \) are the common clauses between \( f \) and \( \phi(f) \);
- the set of literals \( \{l_{j_1}, \ldots, l_{j_{rm}}\} \) appears as fixed by \( \Sigma_2 \) and is made up of:
  
  1. all variables \( \Sigma_2 \) and \( \Sigma_1 \) disagree on, as fixed by \( \Sigma_2 \),
  
  2. all variables \( \Sigma_1 \) and \( \Sigma_2 \) agree on that are not present in the common clauses of \( f \) and \( \phi(f) \),
  
  3. all variables \( \Sigma_1 \) and \( \Sigma_2 \) agree on that allow for \( s_{j_r} \) to be extended without absorbing any other invariant clause.

Once again, it is remarked that there is only one possible form of immediate child of \( f \) in a distinct signature and that this is again a consequence of the fact that the structure of \( f \) cannot be preserved integrally when changing signatures due to variable sign disagreements. The uniqueness of the proposed form also follows from a maximality argument, but this time, it is maximal in the sense that no more clauses can be added to the proposed form without sacrificing immediacy of the proposed child.

**Example 6.2.** Recall example 6.1. Take \( \Sigma_1 \) to be the signature that fixes all variables in their non-complemented version and \( \Sigma_2 \) to be the signature disagreeing with \( \Sigma_1 \) on \( x_2, x_3 \) and \( x_7 \). This example explores the immediate children of \( f' = (x_1 \land x_4) \lor (x_1 \land x_5) \lor (x_1 \land x_6) \lor (x_1 \land x_7) \) in \( \Sigma_1 \). Note that \( f' \) is an element of \( \Sigma_2 \).

The first three clauses in \( f' \) only contain variables \( \Sigma_1 \) and \( \Sigma_2 \) agree on, therefore they are common clauses between \( f' \) and \( \phi(f') \). The same does not apply to the last clause in \( f' \), thus it must be excluded from the constructed immediate children of \( f' \) in \( \Sigma_1 \), according to theorem 6.4.

Clause \((x_1 \land x_4)\) can be extended with \( x_2, x_3, x_6 \), thus it is not necessarily the case that \( f' \) is an immediate child of \( f \) in some signature other than the one \( f \) belongs to found by theorem 6.2; it is not necessarily the case that \( f \) is an immediate child of \( f' \) in the signature of \( f \) found by theorem 6.4 and vice-versa. This is of course not a contradiction, as the setting is as described in Section 4. More formally, let \( f \in \Sigma_1 \) and \( f' \in \Sigma_2 \). \( f' \) being an immediate parent of \( f \) in \( \Sigma_2 \), only ensures no other element of \( \Sigma_2 \) lays between \( f \) and \( f' \). It does not forbid some element of \( \Sigma_1 \) distinct from \( f \) to lay between \( f \) and \( f' \). This is precisely why even when answering the simplified problem, some care must be taken to correctly answer the more general problem as in Section 3.
7. Conclusions and Final Remarks

Imbuing $\mathcal{F}_n$ with the partial order $\preceq$ in definition 2.3, one can ask themselves: given some function $f \in \mathcal{F}_n$, what are the closest functions to $f$ with respect to the relationship given in definition 2.3? To answer this question, the approach undertaken was to add a biologically motivated layer of restrictions that simplifies the problem. The addition of this new layer of restrictions offers a dual advantage: it has contextual meaning and it is mathematically more manageable. The thesis is then dedicated to answering the simplified problem. The broader problem can be answered from the simplified problem at the cost of some extra computational resources.

It is believed that the simplified problem is correctly answered with the newly proposed theorems 5.1 and 5.2, in the case of searching for the closest functions within the same signature as the given function, and with the newly proposed theorems 6.2 and 6.4, in the case of searching for the closest functions within a distinct but fixed signature. In the latter scenario, criteria are also established to decide a priori if the given function has any related functions in that signature or not as established by definition 2.3. It is remarked that it is debatable whether answering the larger problem irregardless of signature considerations is of any significant practical advantage for the motivating biological framework. However, from a mathematical standpoint, it would be quite the achievement.

References


